A SYSTEM FOR COMMUNICATION BETWEEN PRIVATE AND PUBLIC IP NETWORKS

The present invention relates to a system for communication between a first computer terminal in a private IP network and a second computer terminal in a public IP network.

Generally speaking, one particularly advantageous application of the invention is to IP communication between public networks and private networks, especially when a public IP network has to communicate with a private IP network.

RFC (Request For Comments) 1918 of the IANA (Internet Assigned Number Authority) covers private networks and private addresses intended to remedy the limited number of IP addresses in version 4 of the IP standard. Routers of the public network, for example the Internet, cannot route to private addresses, which are assigned in particular ranges. This enables a large number of computer terminals to be connected to the same private network.

Terminals in the private network that need to communicate with computer terminals in the public network must do so via network boundary equipments, some of which are referred to as gateways and have an IP address in the public network and an IP address in the private network. They serve as agents of the terminals of the private network in respect of requests to the public network. They receive requests from a private terminal at their private IP address and forward those requests on the public network using their public IP address. When the response to a request reaches the gateway, the gateway forwards it to the private network terminal that submitted the request using its private address. RFC 3022 covers this NAT (Network Address Translation) mechanism.

Another kind of network boundary equipment associated with gateways and known as a firewall serves

as a security entity controlling access to and from the Internet. Firewalls are the subject of more or less restrictive rules.

Certain types of application use proxies (proximity agents) specific to particular protocols, the best known being HTTP (HyperText Transfer Protocol) and FTP (File Transfer Protocol) proxies. These proxies receive requests from private network terminals and forward them over the public network in the name of the private network terminals. They may be placed behind gateways. Because they are obligatory points for communications conforming to a given protocol to pass through, particular services have been added to them, such as a cache mechanism in the case of HTTP proxies.

One of the main features of the NAT mechanism is that it is asymmetrical. An IP packet can pass freely from the private network to the public network. In contrast, a packet can pass from the public network to the private network only if a packet has previously taken the opposite route. Thus a private network terminal must take the initiative for any communication.

The underlying mechanism is based on the route concept. For an equipment performing a NAT operation, a route is a set of three "IP address-port" pairs. When a packet from the private network reaches the network boundary equipment, which stores the "IP address-port" pair of the computer terminal of the private network that sent the packet, that pair being called pair1. The destination "IP address-port" pair in the public network is called pair2 and the "IP address-port" pair of the public interface of the boundary network equipment via which the packet will be forwarded is called pair3. When a packet from the public network reaches the public interface, the network boundary equipment looks for a corresponding route, i.e. the route whose source corresponds to pair2 and whose destination corresponds to pair3 of a route previously elected. If there is a

previously elected route, the packet is forwarded to pair1 of the elected route. If there is no previously elected route, the packet is not forwarded on the private network. Thus a packet arriving from the public network can be forwarded on the private network only if a packet from the private network has previously created a route for it. Hence the asymmetrical nature of the NAT mechanism.

This mechanism has many limitations that have become increasingly obvious as the diversity of applications on the Internet has increased and private networks have proliferated: private networks are no longer restricted to corporations, but are equally relevant to a large proportion of the general public, usually connected via ADSL.

Consider a help service, for example. Increasing numbers of computer terminals are now equipped with an onboard HTTP server used for the purposes of configuration. For example, it is not possible at present to obtain assistance by having a technician on a public network verify and correct a faulty equipment configuration via an HTTP connection. At best, a technician can ask the customer to redirect a port of a gateway to the equipment in question by the customer personally creating a route, an operation that is just as unfeasible for the customer as it would be for the customer to correct the configuration without help from the technician.

The rules governing firewalls range from the simplest to the most complicated, the simplest rules authorizing everything or nothing. To implement a more coherent security strategy, firewalls at present rely on application type. Authorizing an application to access the Internet in fact authorizes packets addressed to the port associated with that type of application to access the Internet (port 80 for HTTP, port 21 for FTP). Certain new applications, for example games, telephone,

and videophone applications, use a multitude of ports, which are often assigned dynamically. Dynamic assignment prevents the definition of adequate rules. It becomes essential to allow all packets to pass or at best to open a complete range of ports, which is an unsatisfactory strategy from the security point of view.

Accordingly, the technical problem to be solved by the subject matter of the present invention is that of proposing a system for communication between a first computer terminal in a private IP network and a second computer terminal in a public IP network, which communications system includes a network boundary equipment, solves the problem of incoming connections to a private network, simplifies the security strategy applied at the boundary of the private network, without compromising it, and requires minimal or no configuration of the existing elements of the boundary equipment, gateway, and firewall.

According to the present invention, the solution to the stated technical problem consists in said communications system further including a mediation system in the private IP network that is associated with said first terminal and is adapted to make an IP interface available to said second terminal and a control server in the public IP network that is able to control said mediation system via a communications tunnel through said network boundary equipment.

The following description, given with reference to the appended drawing and by way of non-limiting example, explains in what the invention consists and how it may be reduced to practice.

Figure 1 is a general block diagram of a communications system of the invention.

Figure 2 is a detailed diagram of the Figure 1 communications system.

Figure 1 shows a system for communication between a first computer terminal 1 in a private IP network 7 and a

second computer terminal 5 in a public IP network, said system comprising a network boundary equipment 3 of the gateway and/or firewall type. The Figure 1 communications system must be able to communicate through the network boundary equipment 3, simplify the process of configuring the equipment and ensure good performance. The system is independent of the protocols above the TCP (Transmission Control Protocol) and the UDP (User Datagram Protocol). It can be used for all types of application that seek an effective solution to the problem of communicating through gateways and firewalls.

A mediation system 2 on the private network 7 has a TCP/UDP/IP interface and, generally speaking, the terminal 5 must be able to use the TCP/UDP/IP interface of the mediation system 2 to render the service that it offers. The mediation system 2 therefore makes its TCP/UDP/IP interface available to the terminal 5 via a control server 4.

To this end a communications tunnel 6 is created between the mediation system 2 and the control server 4 through the gateway/firewall 3.

The mediation system 2 and the control server 4 are completely generic and only the computer terminal requires the intelligence needed for the service rendered to the terminal 1 of the private network 7.

The internal IP terminal 1 uses the mediation system 2 to communicate with the external IP terminal 5, which communicates with the control server 4. To comply with the general rules for communicating through the NAT mechanism and firewall, the system 2 sets up the tunnel 6 to the server 4 via a single fixed port. This minimizes the impact on the configuration of the router and manages the connections of clients to servers efficiently.

The single fixed port of the control server 4 is referred to below as the "service port".

The tunnel 6 consists of a TCP channel and where applicable a plurality of UDP channels between any port of the system 2 and the service port of the server 4.

A detailed description of the operation of the communications system of the invention is given below with reference to Figure 2.

To be initialized, the mediation system 2 connects to a fixed port of the control server 4 via a TCP channel. It uses this connection to inform the server 4 about its state and about its environment. This information can range from a description of its configuration in the private network 7 (IP address, subnetwork mask, etc.), through authentication or identification, to a description of the service that it wishes to use.

Once initialization has been effected, three types of operation are effected between the system 2 and the server 4:

· Requests: open, redirect, connect, make server, and close ports.

· Packet and event relay.

· Maintain channel.

A light protocol is used between the system 2 and the server 4 to announce and describe these operations. Only the semantics of this protocol are described here, the syntax being left to the discretion of the producers of the service.

1. <u>Open, redirect, and close port requests</u>

The server 4 sends open, redirect and close port requests to the mediation system 2.

· Open port:

The server 4 requests the opening of a port by sending the IP address and the number of the port to be opened (the mediation system 2 may be on a machine of the local area network that has a plurality of IP addresses on that network), the type of service (TCP or UDP), and

the importance assigned to the port number (it is possible to request either a desired (but not obligatory) port number, in which case the first free port starting from the number requested is assigned, or on the contrary an obligatory port number can be requested).

The response to the open port request consists in sending an identifier of the opened port and the assigned port number, or an error code if the request fails.

· Redirect port:

The redirection of ports responds to a particular constraint.

The service channel is based on the TCP and can forward TCP or UDP packets that arrive at the internal interface of the mediation system 2.

Particular characteristics are added to packets in transit arriving over UDP on the TCP channel. The TCP is a reliable protocol compared to the UDP, in that the receiver acknowledges packets in transit over TCP. The sender sends again packets that are not acknowledged. This reliability mechanism involves a delay. Certain applications opt to use the UDP to avoid the delay induced by the reliability mechanism. The object of the port redirection mechanism is to enable the application implementing the invention to retain the UDP as the underlying transport layer.

The server 4 requests the redirection of a port by sending the IP address and the number of the port to be redirected (the mediation system 2 can be on a machine of the local area network that has a plurality of IP addresses on that network), and the importance assigned to the port number (it is possible to request either a desired (but not obligatory) port number, in which case the first free port starting from the number requested is assigned, or on the contrary an obligatory port number can be required).

The response to the redirection request consists in sending an identifier of the redirected port and the

assigned port number, or an error code if the request fails.

Once redirection has been effected, any packet arriving at the redirected port is systematically relayed to the server 4 at its fixed port using the UDP.

· Connect port:

The server 4 requests the mediation system 2 to connect a previously opened TCP port to an IP address and an IP port of the private network 7. The request is sent with the identifier of the previously opened port, the IP address and the port to which it must connect.

The response to the connection request consists in sending an acknowledgement code or an error code if the request fails.

· Make server port:

There are two types of TCP port:

· server (service) ports, which accept connections from other ports and are therefore "listening" (this is typically the case of the ports 80 of HTTP servers), and

· client (connection) ports, which are connected to the server ports.

When opened (using the mechanism described above), a TCP port is not yet dedicated. It becomes a client port (mediation system 2) as soon as the server 4 requests its connection to another port.

To dedicate an open TCP port as a server port, the server portion of the invention sends the client portion (mediation system 2) a make server request giving the identifier of the opened port concerned.

The response to the make server request consists in sending an acknowledgement code, or an error code if the request fails.

· Close port:

The server 4 requests the closing of a port by sending the identifier of the port received at the time of opening or redirecting it.

The response to the close request consists in sending an acknowledgement code, or an error code if the request fails.

2. Packet and event relay

· Packet relay:

This operation is bidirectional. The server 4 can request the mediation system 2 to forward a packet on the private network 7, specifying the identifier of the port to be used to forward the packet (this is the identifier received on opening the port), the IP address and the number of the destination port and the packet to be forwarded.

In the other direction, the mediation system 2 relays a packet that it has received to a port opened beforehand by the server 4, indicating the identifier of the receiver port, the IP address and the number of the sending port and the packet received.

· Event relay:

The events relayed by the client (mediation system 2) to the server 4 are events at the opened ports. These events are:

· Connection of an IP system of the internal network 7 to an opened TCP server port (corresponding to the sequence SYN, SYN/ACK, ACK).

· Request for closure (message TCP FIN) or destruction (message TCP RST) of a TCP connection.

3. Maintain channel

There may be active route scrutinizing mechanisms in all of the network equipments through which the tunnel 6 between the client (system 2) and the server 4 passes. These mechanisms verify that the routes, i.e. the three "IP address-port" pairs, are not obsolescent. If no packet having these pairs for its source and destination coordinates passes through the equipment 3 during a period called the time to live (TTL), the route is

destroyed, for example to prevent a later packet from the server 4 passing through the equipment 3 to the client 2. This breaks the tunnel 6.

To avoid this problem, the client 2 sends a packet (known as the maintain channel packet) over the opened channels to the server 4 before the end of the TTL of the channel in question.